JOIV : Int. J. Inform. Visualization, 8(2) - May 2024 784-793

INTERNATIONAL JOURNAL
ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Implementation of Multi Extension in Blockchain-Based IoT Platform for Industrial IoT Devices

Agus Prayudi [a], Sritrusta Sukaridhoto [b,*], Muhammad Udin Harun Al Rasyid [b], Oktafian Sultan Hakim [b], Yohanes Yohanie Fridelin Panduman [c], Rizqi Putri Nourma Budiarti [d]

[a] *Department of Electrical Engineering, Politeknik Elektronika Negeri Surabaya, Sukolilo, Surabaya, Indonesia*
[b] *Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Sukolilo, Surabaya, Indonesia*
[c] *Graduate School of Natural Science and Technology, Okayama University, Okayama, Japan*
[d] *Department of Information System, Universitas Nadhlatul Ulama Surabaya, Wonocolo, Surabaya, Indonesia*
*Corresponding author: *dhoto.pens.ac.id*

***Abstract*—The rise of the Internet of Things (IoT) has led to the creation of technologies to improve human life. IoT involves integrating the Internet with the physical world, spanning applications like smart homes, industries, supply chains, academia, and more. By the end of 2020, around 212 billion IoT devices were globally deployed, presenting substantial opportunities for manufacturers and diverse applications. There have been numerous implementations of IoT across various fields, including Blockchain IoT (B-IoT), Artificial Intelligence of Things (AIoT), Digital Twin, and new communication protocols like the Matter protocol. We conducted a comprehensive testing of the blockchain (B-IoT) extension system on various bandwidths and scenarios, such as blockchain API execution time, speed, retention performance, and smart contract vulnerability testing. Our testing has been successful, and several messaging systems were used. Kafka was recommended to overcome the pending transaction problem caused by unprocessed messages. Our smart contract exhibited high severity. The Artificial Intelligence of Things extension, tested on real environments for person and vehicle counters, has shown successful results. Digital Twin, integrated into the IoT platform to perform and control 3D assets such as the postgraduate PENS building, has demonstrated efficient performance. Matter protocol achieved an average task execution speed of 0.48 tasks per second. Matter P2P communication was also successfully tested in this research by implementing the Access Control List (ACL) command.**

***Keywords*— Blockchain; artificial intelligence of things; digital twin; matter protocol.**

Manuscript received 7 Dec. 2023; revised 6 Jan. 2024; accepted 3 Feb. 2024. Date of publication 31 May 2024.
International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. Introduction

The rise of the Internet of Things (IoT) has spurred the development of technologies to enhance human life. Essentially, IoT represents the merging of the Internet with the physical world, encompassing applications like smart homes, industries, supply chains, academics, and more [1], [2]. At the end of 2020, there will be approximately 212 billion globally deployed IoT entities, offering significant opportunities for manufacturers and various applications [3]. Currently, there are many implementations of IoT across various fields, such as Blockchain IoT (B-IoT), Artificial Intelligence of Things (AIoT), Digital Twins, and new communication protocols like the Matter protocol.

Related to the blockchain, most IoT implementations currently follow a centralized paradigm, requiring centralized access management and conventional data storage [4]. However, security concerns have led to recent attempts at integrating blockchain and IoT [5]–[7]. These endeavors aim to address security issues in IoT by leveraging blockchain's immutability and decentralized consensus mechanism.

Many IoT platforms continue to face a deficiency in essential security technologies, especially data aggregation [8]. In IoT, a thorough security solution is imperative, particularly for critical applications where sensitive data is stored, or a high degree of trust is necessary [9]. Blockchain technology has been implemented in several fields, such as the Internet of Things, supply chain, etc. Many researchers, such as [10]–[12], have researched the implementation of blockchain IoT in several fields, such as healthcare and industry.

Numerous blockchain implementations exist for managing system websites like [13] where they integrated Hyperledger Fabric into OpenEMR, a hospital management system application. Their system uses Change Data Capture with the Debezium connector and Kafka as the message broker. Whenever data in the database changes, Debezium automatically records and stores it in the Kafka topic. Misra et al. [14] have utilized blockchain at the edge to enhance the security of IoT applications. They implemented the Ethereum blockchain and investigated network latency impacts, scaling blockchain nodes, and varying data sizes. Recently, blockchain has found its way to integrate with another emerging technology, namely mixed reality. Arissabarno et al. [15] have integrated blockchain with mixed reality to control a smart lab IoT system.

On the other hand, Artificial Intelligence (AI) and the Internet of Things (IoT) have become increasingly prevalent across various facets of our everyday existence [16]. There were several ways to build models for data processing using artificial intelligence algorithms, such as "Abstracted AI" which uses machine learning to train models on the ready data sets, "Centralized Approach," send data to the cloud to train the model, and "Edge Approach" will process the data on the edge computing and send less data to the cloud [17].

Currently, Container technology provides a streamlined cloud environment that enables the deployment and execution of applications in an isolated manner, offering a higher application density on a single host compared to virtual machines [18]. Utilizing advanced machine learning and deep learning on IoT data facilitates real-time predictions and delivers advanced analytics. Integrating IoT and AI at the edge level supports various applications, including image processing, face recognition, video processing, object segmentation, and object tracking. Based on this study, we built an AIoT extension in the IoT platform to enable users to process their data using artificial intelligence.

Related to Digital Twins (DT), The combination of Digital Twin and blockchain technologies within an IoT platform presents a cutting-edge and holistic approach for smart buildings, guaranteeing the smooth incorporation of a 3D system and robust data security via blockchain [19]. The virtual representation of the physical building provided by Digital Twin, coupled with real-time monitoring and predictive features, empowers building managers to enhance operational efficiency, energy conservation, and maintenance activities [20].

IoT platforms have been developed for monitoring and managing IoT devices using various protocols like MQTT, HTTP, Kafka, and more [21]–[26]. For digital twins' implementation. Falah et al. [27] discussed the design system for cyber-physical system implementation using a conveyor. The design system of the IoT platform for cyber-physical system implementation has been studied by [28]. Based on these related works, our uniqueness is implementing Matter protocol as a communication protocol between cyber and physical. In IoT platform studies, we found that there is still a lack of integration extension for integrating IoT platforms with emerging technology.

Currently, only a few research papers have discussed Matter Protocol. Zegeye et al. address the heterogeneity problem in smart homes by utilizing the Matter protocol [29]. Sukaridhoto et al. [30] also have been developing IoT concepts using Matter for healthcare monitoring based on the Internet of Things. Based on that research, our uniqueness is the implementation of Matter protocol in the industrial Internet of Things. This paper utilizes Matter as a communication protocol for controlling the conveyor.

Our proposed system introduces a multi-extension. This extension enhances the capabilities of the IoT platform by enabling seamless integration with blockchain technology, AI application, Matter protocol, and Digital Twin (DT). These extensions were designed to work with IoT platforms and will be accessible through a community repository for the convenience of Home Assistant users. Our contribution is multi-extension architecture and implementation, which allows us to:

- Providing a docker-compose for Blockchain IoT (B-IoT) development configuration to run blockchain networks. This eliminates the number of virtual machines and simplifies the setup process for users.
- Artificial Intelligence of Things (AIoT) has been implemented as an extension (Centralized Approach). Currently, we propose this extension for person and vehicle counting. The extension enables us to get data from the IoT platform by subscribing to the entity data from camera devices.
- Integrating IoT platform with Digital Twin (DT) 3D assets to enable intuitive control and monitoring of IoT activity.
- Implementing and studying Matter protocol extension performance and Matter P2P communication.

II. Materials and Method

This section describes our proposed system in detail, including testing scenarios for blockchain connectors, Matter protocol extensions, Digital Twin extensions, and AI extensions. The details of the extension will be figured out in the next section. The overall design system was presented in Fig. 1. The proposed Blockchain extension implementation has been figured in Fig. 2. Proposed Digital Twin (DT) extension in Fig. 3. Proposed Matter protocol extension in Fig. 4. Proposed AI (Artificial Intelligence) extension in Fig. 5.

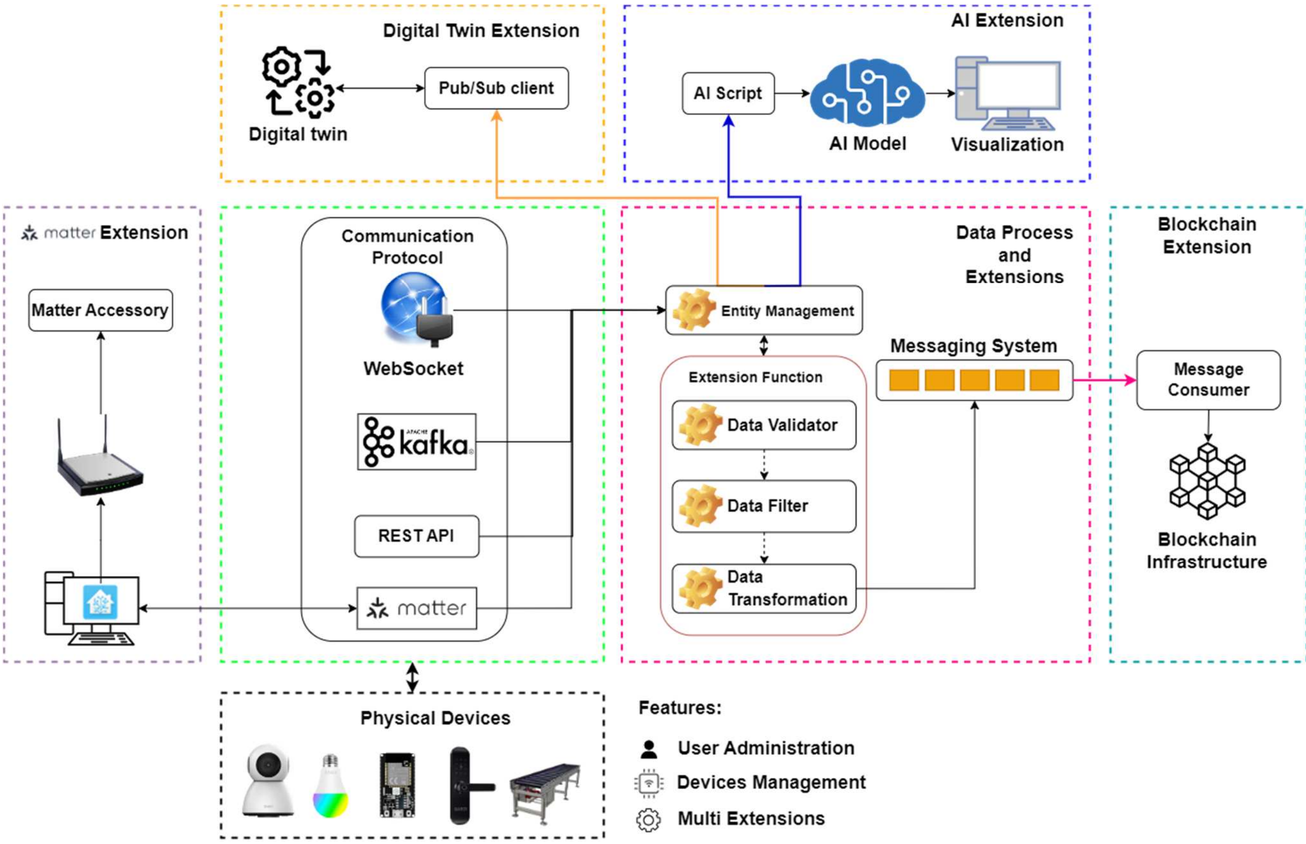


Fig. 1 Overall System of Implementation of Multi Extension in IoT Platform

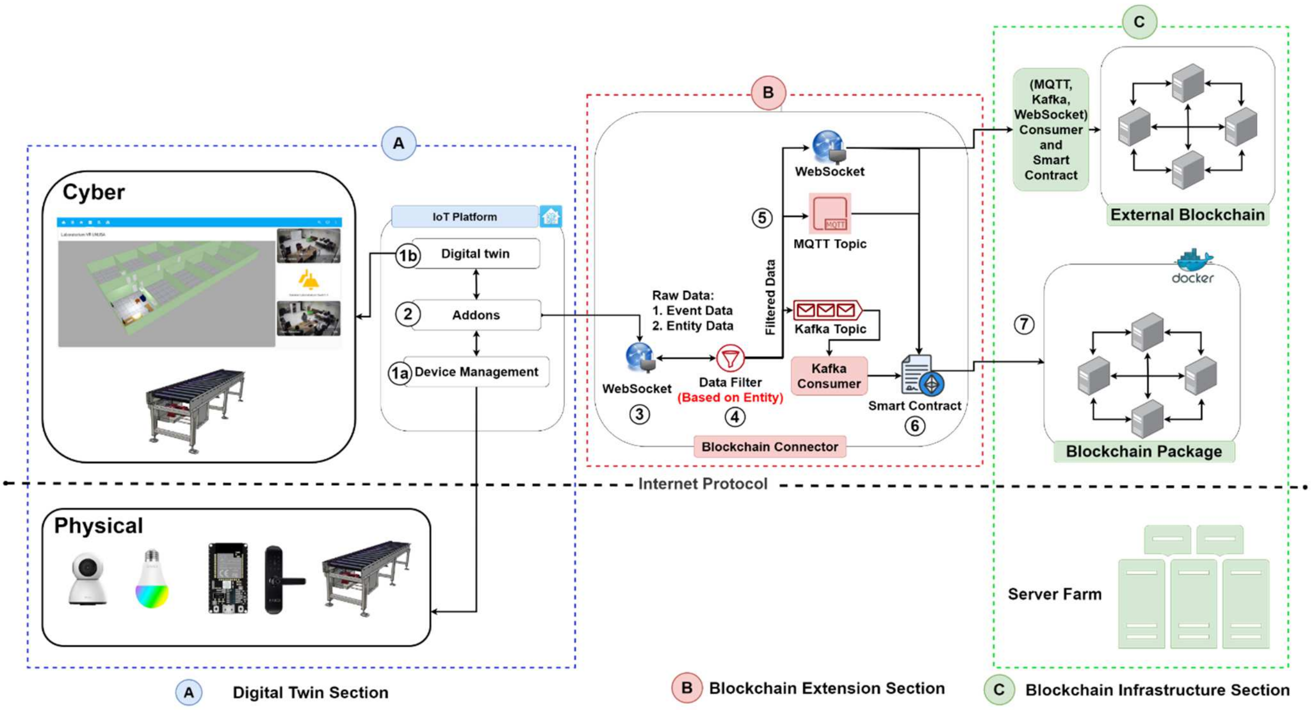


Fig. 2 Implementation of Blockchain Extension

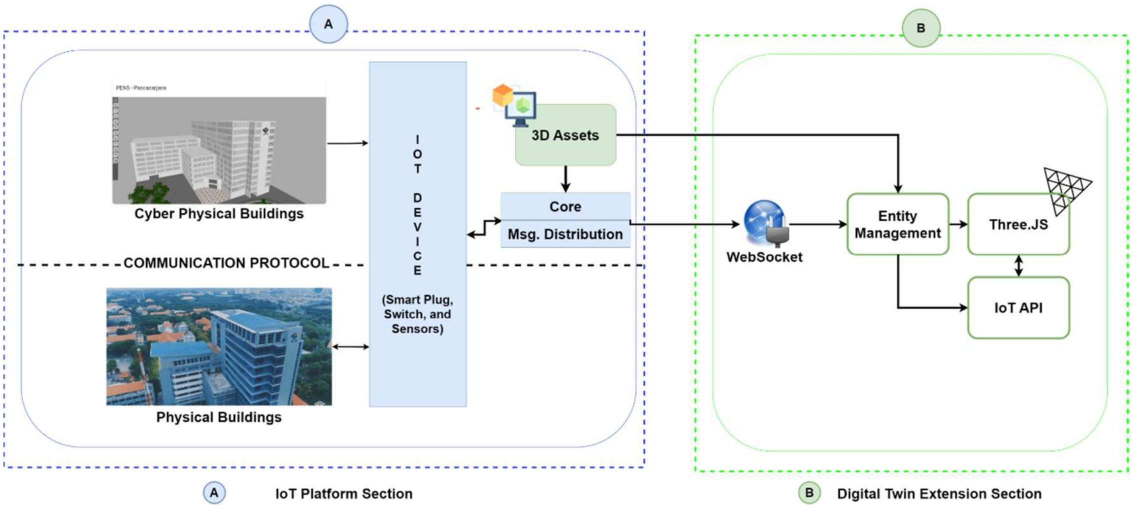


Fig. 3 Implementation of Digital Twin Extension in IoT Platform

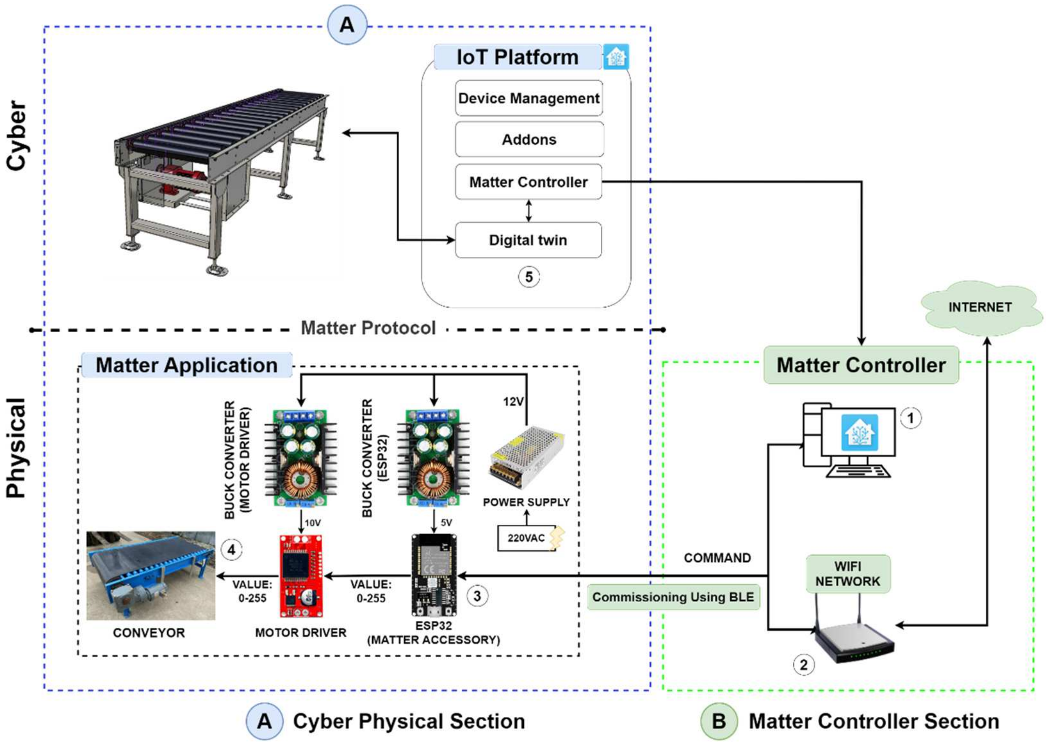


Fig. 4 Design and Implementation Matter Protocol Extension in IoT Platform

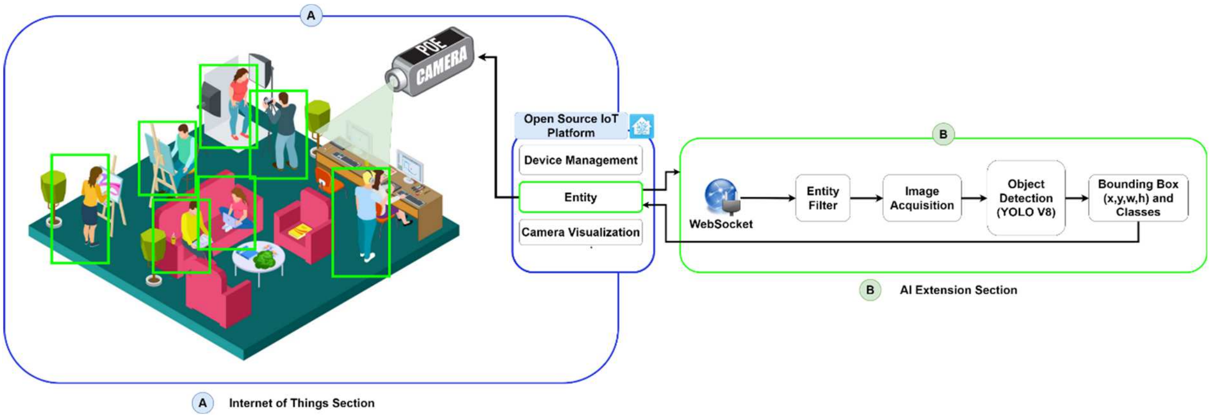


Fig. 5 Implementation of Artificial Intelligence Extension

A. System Architecture

Fig. 1 shows the overall system of the IoT platform that implements multi-extension. Based on Fig. 1The resulting study is in Table I, where we break down the system into several sections based on extension functionality. Detailed information on each section inside the proposed system is explained below.

TABLE I
INTERNET OF THINGS PLATFORM FEATURES COMPARISON

Platform	Parameters					
	Things	**Connectivity**	**AI**	**Blockchain**	**Digital Twin**	**Matter Protocol**
Tuya	All Devices	HTTP, MQTT	X	X	X	X
Things Board	All Devices	HTTP, MQTT, CoAP	X	X	X	X
Azure IoT	All Devices	HTTP, MQTT, AMPQ	✓	X	X	X
SEMAR	All Devices	HTTP, MQTT	✓	X	X	X
OpenHAB	All Devices	HTTP, MQTT	X	X	X	X
Home Assistant	All Devices	HTTP, MQTT, Matter	X	X	X	✓
Blynk	All Devices	Binary Protocol	X	X	X	X
Proposed System	All Devices	HTTP, MQTT, Kafka, Matter, WebSocket	✓	✓	✓	✓

B. Blockchain Extension Implementation

In this section, we discussed all the parts of blockchain extension in detail. By referencing Fig. 1, we break down that system design into several points with specific explanations. Blockchain extension was depicted Fig. 2. The data processing and storing procedure of our blockchain extension is illustrated on Algorithm 1.

Algorithm 1. Blockchain Extension Algorithm

Data: raw data of IoT devices
Result: filtered data of IoT devices

```
expected entity ← ["light", "switch", "boolean"];
while true do:
    subscribing IoT devices data;
    read entity data;
    if entity == expected entity then
        reformatting data by merging the new event
        and old event data;
        return filtered data;
        if filtered data == true then
            store to the messaging system (WebSocket,
            MQTT, Kafka);
            subscribe data from the messaging system;
            trigger smart contract to store to the
            blockchain;
    else
        nothing to do and back to looping;
```

We subscribed to IoT data based on their entity using a generated user token, utilizing WebSocket API. Our extension filters the data using the entity's name and formats it into the new data structure. We provided Kafka, MQTT, and WebSocket as a messaging system. By adding a messaging system, the user could build their smart contract. Smart contracts were created using the Solidity programming language. A smart contract is used for the Ethereum blockchain network. Furthermore, we compiled the smart contract into ABI JSON format. The ABI JSON will create the API (Application Programming Interface) or function using Web3JS. The smart contract API algorithm is shown in Algorithm 2.

Algorithm 2. Smart Contract Algorithm

Data: privateKey, IoT Data

```
Struct {
    string IoT Data;
    string date;
    address payable user;
}
Function History(IoT Data, date, user){
    count++;
    entityState[count]←EntityState(IoT
     Data, date, user);
    Result: response(error, block hash, block number,
     transaction hash, gas used, status)
}
While true do
    get account private key;
    if (privateKey !== null) then
        encode ABI from smart contract;
        create transaction consisting (nonce,
        data, contract address);
        sign transaction data;
        if data is signed then
            call History function;
            return response (error, block hash, block
            number, transaction hash, gas used, status);
```

By utilizing Hyperledger Besu, we have used QBFT Proof of Authority (PoA) consensus protocol. Adding a block to the chain requires the majority consensus (equal to or higher than 2/3) of the validators and nodes responsible for validating transactions. The overall blockchain infrastructure is illustrated in Fig. 2, in the blockchain infrastructure section. Based on this study, we proposed the four blockchain nodes using docker-compose so that users can install the blockchain by installing the docker and running the predefined docker-compose. The predefined docker-compose consists of a blockchain network and volume configuration to bind the docker with the local environment.

C. IoT Platform and Blockchain Connector Specification

Table II describes the IoT platform testing environment, and Table III describes the blockchain testing environment. The environment specifications on those tables were utilized for multi-extension deployment and testing.

TABLE II
ENVIRONMENT TESTING FOR DIGITAL TWIN, MATTER, AND AI EXTENSION

Component	Specification
RAM	4GB
Hard disk	32GB
CPU	4CPU(s), 91MHz used
Virtual Machine	1 VM
Software Version	Home Assistant OS v2023.5.2, YOLOv8, Espressif SDK for Matter protocol
Programming Language	Solidity v0.8, Node JS v.18.2.1, Python3

In this research, we optimized the usage of virtual machines by implementing blockchain using docker-compose. Table III presents the specifications for blockchain extension and infrastructure.

TABLE III
ENVIRONMENT TESTING FOR BLOCKCHAIN EXTENSION

Component	Specification
RAM	8 GB
Hard disk	150 GB
CPU	4 CPU(s), 1382 MHz used
Virtual Machine	1 VM 4 Container
Software Version	Hyperledger Besu 22.7.4, Docker Compose v1.25.0, Quorum Block Explorer
Programming Language	Yaml

D. Digital Twin Extension Implementation

Digital Twin extension was implemented in the IoT platform by utilizing sweet home software for building the 3D object of smart building. We integrated IoT devices such as lamps, door locks, and cameras into the 3D object by connecting the entity name of IoT devices to the 3D object. Digital Twin Extension was depicted in Fig. 3.

E. Matter Protocol Extension Implementation

Matter protocol has been implemented as communication protocol. We can create our own devices using Matter firmware. In this research, the Matter protocol will be modified to fulfill our needs to test the performance. The proposed Matter extension was depicted in Fig. 4.

F. Artificial Intelligence Extension Implementation

In this research, we developed an AI extension for computer vision applications. This extension can detect people in the room and vehicles in the parking space area. We used YOLO V8 as a computer vision model and GPU 6000 RTX for AI extension processing in the AI extension. The proposed design system for implementing the AI extension is depicted in Fig. 5.

III. RESULTS AND DISCUSSION

There are various tests for blockchain extension, such as blockchain API execution time testing, block speed, retention performance, and smart contract vulnerability testing. For the Matter protocol extension, we have tested the performance of Matter task execution speed and P2P communication based on the Matter protocol. For the Digital Twin extension, we tested the load performance of 3D assets in the digital twin extension. For AI extension, we tested the system in a real environment to ensure that our system is working successfully.

A. Blockchain API Execution Time on Different Messaging Systems

The result of blockchain API time characteristic is shown in Fig. 6, Fig. 7, and Fig. 8.

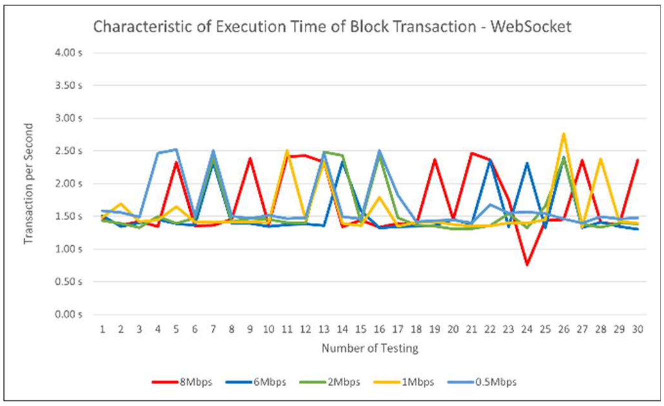


Fig. 6 Blockchain API Execution Time in Various Bandwidths Using WebSocket

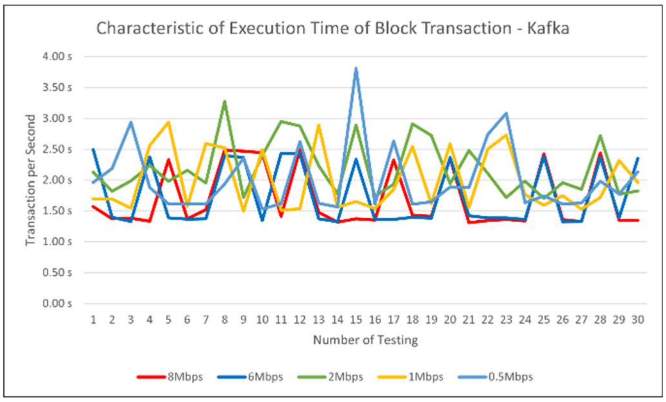


Fig. 7 Blockchain API Execution Time in Various Bandwidths Using Kafka

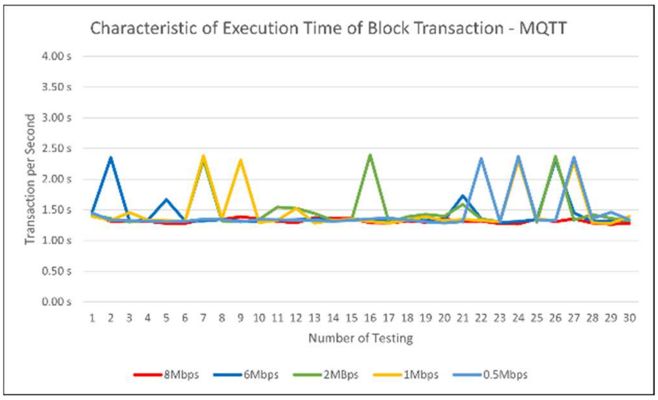


Fig. 8 Blockchain API Execution Time in Various Bandwidths Using MQTT

In this testing, we conducted blockchain API (Application Programming Interface) execution time testing by utilizing several bandwidths (0.5 Mbps, 1 Mbps, 2 Mbps, 6 Mbps, and

8 Mbps). Based on Fig. 6, the blockchain API (Application Programming Interface) execution time characteristics fluctuate. Based on Fig. 7, the characteristics are still fluctuating, the graph tends to have a high execution time, it is because Kafka has more built-in processing for queuing the data.Based on Fig. 8, the blockchain API execution time characteristics fluctuate less.

B. Block Speed and Retention Performance on Different Messaging Systems

In Table IV, WebSocket section, our system can store the data to the blockchain 100% in every tested bandwidth. In WebSocket testing, we got the highest Transaction Speed at 6 Mbps with a value of 0.64 TPS. WebSocket was implemented by using a WebSocket server built into the IoT platform itself.

TABLE IV
BLOCKCHAIN EXTENSION PERFORMANCE USING WEB SOCKET

Bandwidths (Mbps)	**Transaction per Second (TPS) WebSocket**				
	8	**6**	**2**	**1**	**0.5**
Avg. Time (seconds)	1.71	1.54	1.58	1.58	1.67
TPS	0.59	0.64	0.63	0.63	0.6
Retention (%)	100	100	100	100	100
Avg. TPS			0.62		

Based on Table V, in the MQTT (Message Queuing Telemetry Transport) section, our system can store the data 100% to the blockchain networks. We got the highest transaction Speed Per Second at 8 MBps with 0.76 TPS and the smallest value at 2 Mbps with 0.66 TPS. We implemented MQTT (Message Queuing Telemetry Transport) as a messaging system to distribute the data to the external client and handle string data.

TABLE V
BLOCKCHAIN EXTENSION PERFORMANCE USING MQTT

Bandwidths (Mbps)	**Transaction per Second (TPS) MQTT**				
	8	**6**	**2**	**1**	**0.5**
Avg. Time (seconds)	1.32	1.43	1.5	1.47	1.44
TPS	0.76	0.7	0.66	0.68	0.69
Retention (%)	100	100	100	100	100
Avg. TPS			0.7		

In Table VI, Kafka section, our system is able to store the data 100% to the blockchain networks in every tested bandwidth. The optimal Transaction Per Second value is 8 Mbps with 0.76 TPS and the smallest TPS value is 0.46 TPS at 2 Mbps.

TABLE VI
BLOCKCHAIN EXTENSION PERFORMANCE USING KAFKA

Bandwidths (Mbps)	**Transaction per Second (TPS) Kafka**				
	8	**6**	**2**	**1**	**0.5**
Avg. Time (seconds)	1.69	1.74	2.19	1.99	2.02
TPS	0.76	0.57	0.46	0.5	0.5
Retention (%)	100	100	100	100	100
Avg. TPS			0.56		

C. Smart Contract Vulnerability Testing

In this section, we also tested our smart contract vulnerability to measure its severity and check the possibility of drawbacks. The testing was conducted by using Mythril Ethereum Smart Contract Vulnerability Tester. The result was depicted in Fig. 9.

```
==== Integer Arithmetic Bugs ====

Severity: High
Contract: HomeAssistant
Function name: createHistory(string,string)
Estimated Gas Usage: 29898 - 136483
The arithmetic operator can overflow.
It is possible to cause an integer overflow or underflow in the arithmetic operation.

In file: /tmp/HomeAssistant.sol:15
```

Fig. 9 Vulnerability Testing Result Using Mythril Ethereum Smart Contract Tester

D. Digital Twin Testing

In this section, we will present the digital twin performance results obtained from testing the digital twin in IoT platform by using 18Mbps bandwidth. We tested the load performance of 3D assets (12th Floor Postgraduate PENS Building). The result of digital twin performance can be seen in Fig. 10.

Fig. 10 Postgraduate PENS Digital Twin System

TABLE VII
DIGITAL TWIN WEB LOAD PERFORMANCE TESTING

Data	**Buildings**	**Size File (MB)**	**Amount of Floor**	**Load Time (s)**
1	Postgraduate PENS	24.5	12th	8.65
2	Postgraduate PENS	24.5	12th	8.87
3	Postgraduate PENS	24.5	12th	8.5
4	Postgraduate PENS	24.5	12th	8.43
5	Postgraduate PENS	24.5	12th	8.26
6	Postgraduate PENS	24.5	12th	8.63
7	Postgraduate PENS	24.5	12th	8.75
8	Postgraduate PENS	24.5	12th	8.55
9	Postgraduate PENS	24.5	12th	8.67
10	Postgraduate PENS	24.5	12th	8.54
Average Load Time (s)				8.58

Based on Table VII, we also visualized the result in chart to see the effect of 3D design file size to the load time. The result was depicted in Fig. 11.

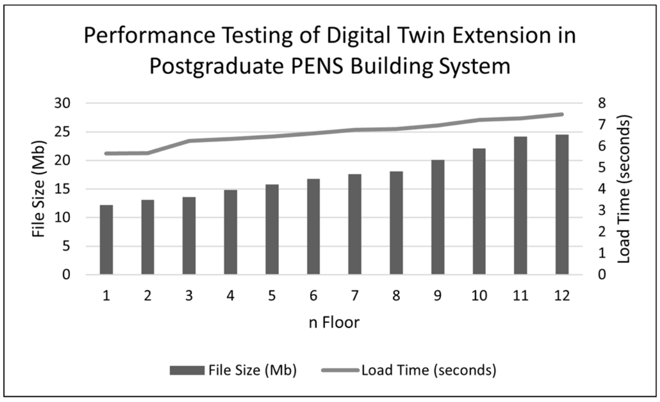


Fig. 11 Performance Testing of Digital Twin Extension on each Floor of the Postgraduate PENS Building System

According to Fig. 11, the smaller the file size of the 3D building of postgraduate PENS, the faster the load time for loading the 3D asset in the IoT platform.

E. Matter Protocol Testing

Matter Task execution Speed testing is conducted on various bandwidths (1 Mbps, 0.5 Mbps, 0.256 Mbps, 0.128 Mbps, 0.064 Mbps). We also tested P2P communication using the Matter protocol. The testing result and scenario are explained below.

1) Matter Task Execution Speed Testing: In this section, we tested the task execution speed by sending a command from Matter controller to the Matter accessory. The task execution speed testing based on bandwidth is shown in Table VIII.

TABLE VIII
MATTER PROTOCOL TASK EXECUTION PERFORMANCE TESTING

Task per Second (TPS) - Matter Protocol					
Bandwidths (Mbps)	1	0.5	0.256	0.128	0.064
Avg. Time (milliseconds)	2079	2088	2092	2094	2097
TPS	0.481	0.479	0.478	0.478	0.477
Avg. TPS			0.48		

Based on Table VIII, we concluded that Matter has good performance even on the small bandwidth (0.064 Mbps), with the execution speed only 4ms different compared to 1 Mbps bandwidth. The result was depicted in Fig. 12.

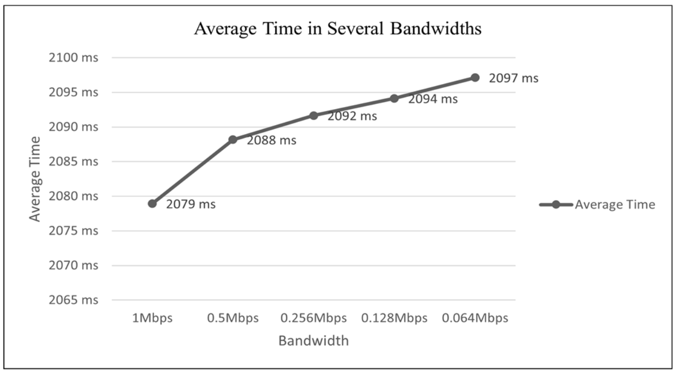
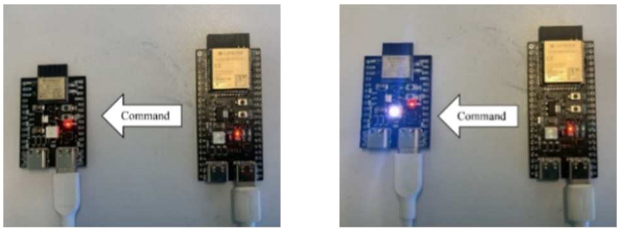


Fig. 12 Characteristic of Average Time of Matter Task Execution in Several Bandwidths

Based on Fig. 12, the average time of Matter execution time has increased inversely and proportionally to bandwidth. The lowest average time was obtained at 1 Mbps, which is 2079 milliseconds, and the highest average time was obtained at 0.064 Mbps which is 2097 milliseconds.

2) Matter Point-to-Point Communication Testing: Matter P2P communication testing is leveraged using ESP32-C3 as a server and ESP32-S3 as a client. The result is shown in Fig. 13.



(a) (b)

Fig. 13 Point-to-Point Communication Testing using Matter Protocol: (a) Before Send Command (b) After Send Command

From this testing, Matter can be used for interconnected devices. P2P communication was successfully tested using the ACL command provided by the Espressif SDK for the Matter protocol.

F. Artificial Intelligence of Things Testing

In this section, we tested the AI extension for object recognition cases, such as person counting and vehicle counting. We tested the system by using a qualitative testing method. The testing was depicted in Fig. 14.

(a)

(b)

Fig. 14 AI Extension Testing: (a) Detected People in The Room Space Using AI Extension, (b) Detected vehicle in the parking space using AI extension

Based Fig. 14, our system is successfully launched our extension by using YOLO v8x model and python library to recognize person and vehicle in the video or image input with mAP (mean Average Precession) is 53.9%.

IV. Conclusions

This paper presented the architecture and implementation of multiple extensions such as Blockchain IoT (B-IoT) extension, Artificial Intelligence of Things (AIoT) extension, Digital Twin (DT) extension, and Matter protocol extension in IoT platform. We have integrated IoT platform with blockchain as an extension. We tested blockchain extension on several scenarios (Blockchain API execution time, Blockchain API speed, Blockchain, retention performance, and Smart Contract Vulnerability Testing). In WebSocket testing, we got the highest Transaction Speed at 6 Mbps with a value of 0.62 TPS. In MQTT (Message Queuing Telemetry Transport) testing, we got the highest transaction Speed Per Second at 8 Mbps with 0.7 TPS. In Kafka, the optimal Transaction Per Second value is 8 Mbps with 0.56 TPS. We also tested smart contract vulnerability. The results are high severity.

The proposed Digital Twin extension was used to control and monitor IoT devices using 3D objects of building. We obtained the average load time performance of the Digital Twin extension, which is 8.85 seconds for a 24.5MB file size. We proposed a matter protocol extension for increasing interoperability through connectivity with industrial IoT devices by implementing P2P communication. The average task execution time for Matter protocol is 0.48 seconds. We provided an AI extension for counting people in the room space and vehicles in a parking space, with a mean Average Precision of 53.9%. In the future, we will continue improving AI extension flexibility to use custom models. Then, we would like to integrate the proposed system with various application systems and use cases.

Acknowledgment

This research was supported by Board of Research and Community Service 2023 by Ministry of Education, Culture, Research, and Technology (Kemdikbudristek) and Politeknik Elektronika Negeri Surabaya. (Grant Numbers: 168/SPK/D.D4/PPK.01.APTV/VI/2023).

References

[1] A. A. Sadawi, M. S. Hassan, and M. Ndiaye, “A Survey on the Integration of Blockchain With IoT to Enhance Performance and Eliminate Challenges,” IEEE Access, vol. 9, pp. 54478–54497, 2021, doi: 10.1109/access.2021.3070555.

[2] T. Kim, C. Ramos, and S. Mohammed, “Smart City and IoT,” Future Generation Computer Systems, vol. 76, pp. 159–162, Nov. 2017, doi:10.1016/j.future.2017.03.034.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347–2376, 2015, doi:10.1109/comst.2015.2444095.

[4] O. Novo, “Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT,” IEEE Internet of Things Journal, vol. 5, no. 2, pp. 1184–1195, Apr. 2018, doi: 10.1109/jiot.2018.2812239.

[5] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, “Blockchain-Based Decentralized Trust Management in Vehicular Networks,” IEEE Internet of Things Journal, vol. 6, no. 2, pp. 1495–1505, Apr. 2019, doi: 10.1109/jiot.2018.2836144.

[6] O. Novo, “Scalable Access Management in IoT Using Blockchain: A Performance Evaluation,” IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4694–4701, Jun. 2019, doi: 10.1109/jiot.2018.2879679.

[7] E. Tijan, S. Aksentijević, K. Ivanić, and M. Jardas, “Blockchain Technology Implementation in Logistics,” Sustainability, vol. 11, no. 4, p. 1185, Feb. 2019, doi: 10.3390/su11041185.

[8] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, “Blockchain for the IoT and industrial IoT: A review,” Internet of Things, vol. 10, p. 100081, Jun. 2020, doi: 10.1016/j.iot.2019.100081.

[9] D. Minoli and B. Occhiogrosso, “Blockchain mechanisms for IoT security,” Internet of Things, vol. 1–2, pp. 1–13, Sep. 2018, doi:10.1016/j.iot.2018.05.002.

[10] D. Miller, “Blockchain and the Internet of Things in the Industrial Sector,” IT Professional, vol. 20, no. 3, pp. 15–18, May 2018, doi: 10.1109/mitp.2018.032501742.

[11] A. Dwivedi, G. Srivastava, S. Dhar, and R. Singh, “A Decentralized Privacy-Preserving Healthcare Blockchain for IoT,” Sensors, vol. 19, no. 2, p. 326, Jan. 2019, doi: 10.3390/s19020326.

[12] L. Tseng, X. Yao, S. Otoum, M. Aloqaily, and Y. Jararweh, “Blockchain-based database in an IoT environment: challenges, opportunities, and analysis,” Cluster Computing, vol. 23, no. 3, pp. 2151–2165, Jul. 2020, doi: 10.1007/s10586-020-03138-7.

[13] Naufal Adi Satrio, Sritrusta Sukaridhoto, M. Udin Harun Al Rasyid, Rizqi Putri Nourma Budiarti, Ilham Achmad Al-Hafidz, and Evianita Dewi Fajrianti, “Blockchain integration for hospital information system management,” Bali Medical Journal, vol. 11, no. 3, pp. 1195–1201, Sep. 2022, doi: 10.15562/bmj.v11i3.3540.

[14] S. Misra, A. Mukherjee, A. Roy, N. Saurabh, Y. Rahulamathavan, and M. Rajarajan, “Blockchain at the Edge: Performance of Resource-Constrained IoT Networks,” IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 1, pp. 174–183, Jan. 2021, doi:10.1109/tpds.2020.3013892.

[15] C. Arissabarno *et al.*, “Blockchain Integration for Mixed Reality Based Smart Lab Systems,” in *2023 International Electronics Symposium (IES) (IES 2023)*, Bali, Indonesia, Aug. 2023, p. 7.

[16] O. Debauche, S. Mahmoudi, S. A. Mahmoudi, P. Manneback, and F. Lebeau, “A new Edge Architecture for AI-IoT services deployment,” Procedia Computer Science, vol. 175, pp. 10–19, 2020, doi:10.1016/j.procs.2020.07.006.

[17] S. B. Calo, M. Touna, D. C. Verma, and A. Cullen, “Edge computing architecture for applying AI to IoT,” 2017 IEEE International Conference on Big Data (Big Data), Dec. 2017, doi:10.1109/bigdata.2017.8258272.

[18] C. Puliafito, E. Mingozzi, F. Longo, A. Puliafito, and O. Rana, “Fog Computing for the Internet of Things,” ACM Transactions on Internet Technology, vol. 19, no. 2, pp. 1–41, Apr. 2019, doi: 10.1145/3301443.

[19] F. Tao, B. Xiao, Q. Qi, J. Cheng, and P. Ji, “Digital twin modeling,” Journal of Manufacturing Systems, vol. 64, pp. 372–389, Jul. 2022, doi: 10.1016/j.jmsy.2022.06.015.

[20] C. Pylianidis, S. Osinga, and I. N. Athanasiadis, “Introducing digital twins to agriculture,” Computers and Electronics in Agriculture, vol. 184, p. 105942, May 2021, doi: 10.1016/j.compag.2020.105942.

[21] Y. Y. Fridelin, M. R. Ulil Albaab, A. R. Anom Besari, S. Sukaridhoto, and A. Tjahjono, “Implementation of Microservice Architectures on SEMAR Extension for Air Quality Monitoring,” 2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), Oct. 2018, doi: 10.1109/kcic.2018.8628575.

[22] Y. Y. F. Panduman, A. R. A. Besari, S. Sukaridhoto, R. P. N. Budiarti, R. W. Sudibyo, and F. Nobuo, “Implementation of Integration VaaMSN and SEMAR for Wide Coverage Air Quality Monitoring,” Telkomnika (Telecommunication Computing Electronics and Control), vol. 16, no. 6, p. 2630, Dec. 2018, doi:10.12928/telkomnika.v16i6.10152.

[23] Y. Y. F. Panduman, N. Funabiki, P. Puspitaningayu, M. Kuribayashi, S. Sukaridhoto, and W.-C. Kao, “Design and Implementation of SEMAR IoT Server Platform with Applications,” Sensors, vol. 22, no. 17, p. 6436, Aug. 2022, doi: 10.3390/s22176436.

[24] Y. Y. F. Panduman et al., “An Edge Device Framework in SEMAR IoT Application Server Platform,” Information, vol. 14, no. 6, p. 312, May 2023, doi: 10.3390/info14060312.

[25] Y. Y. F. Panduman, N. Funabiki, P. Puspitaningayu, M. Sakagami, and S. Sukaridhoto, “Implementations of Integration Functions in IoT Application Server Platform,” 2022 Fifth International Conference on Vocational Education and Electrical Engineering (ICVEE), Sep. 2022, doi: 10.1109/icvee57061.2022.9930422.

[26] Y. Y. F. Panduman, S. Sukaridhoto, A. Tjahjono, and R. P. N. Budiarti, “Implementation SEMAR-IoT-Platform for Vehicle as a Mobile

Sensor Network," JOIV : International Journal on Informatics Visualization, vol. 4, no. 4, pp. 201–207, Dec. 2020, doi:10.30630/joiv.4.4.425.

[27] Y. Y. F. Panduman, S. Sukaridhoto, A. Tjahjono, and R. P. N. Budiarti, "Implementation SEMAR-IoT-Platform for Vehicle as a Mobile Sensor Network," JOIV : International Journal on Informatics Visualization, vol. 4, no. 4, pp. 201–207, Dec. 2020, doi:10.30630/joiv.4.4.425.

[28] Y. Y. Fridelin Panduman, S. Sukaridhoto, and A. Tjahjono, "A Survey of IoT Platform Comparison for Building Cyber-Physical System Architecture," 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Dec. 2019, doi:10.1109/isriti48646.2019.9034650.

[29] W. Zegeye, A. Jemal, and K. Kornegay, "Connected Smart Home over Matter Protocol," 2023 IEEE International Conference on Consumer Electronics (ICCE), Jan. 2023, doi:10.1109/icce56470.2023.10043520.

[30] S. Sukaridhoto, A. Prayudi, M. U. H. Al Rasyid, and R. P. Nourma Budiarti, "A survey and conceptual of Internet of Things system for remote healthcare monitoring," Bali Medical Journal, vol. 12, no. 3, pp. 2840–2845, Sep. 2023, doi: 10.15562/bmj.v12i3.4441.